REMARKS

Claims 1-30 were pending in the application prior to entry of the present amendment.

Claims 1 and 15 are herein amended.

Claims 1-30 were rejected under 35 U.S.C. Section 102(e) as being unpatentable over Spiegel et al. (USPN 5,615,282). These rejections are respectfully traversed based on the following reasoning.

Claim 1 as herein amended recites:

> "An adder tree for adding numbers comprising:
>
> one or more addition levels including a top addition level and a bottom addition level, wherein a summation of said numbers begins at said top level and propagates through said one or more addition levels, wherein each of said addition levels comprises one or more adder cells;
>
> wherein each of said adder cells is configured to receive a first input operand, a second input operand, a first winner-take-all (WTA) bit and a second WTA bit, and to generate a first output operand, wherein the first output operand equals the first input operand if the first WTA bit is high, wherein the first output operand equals to the second input operand if the second WTA bit is high; and
>
> wherein each of said one or more adders at the top addition level receives two of said numbers as the corresponding first input operand and the second input operand."

This combination of features is not taught or suggested by Spiegel et al. (hereinafter referred to simply as "Spiegel"). In particular, Spiegel does not teach or suggest that

> "each of said adder cells is configured to receive a first input operand, a second input operand, a first winner-take-all (WTA) bit and a second WTA bit, and to generate a first output operand, wherein the first output operand equals the first input operand if the first WTA bit is high, wherein the first output operand equals to the second input operand if the second WTA bit is high"

as recited in claim 1.

Spiegel discloses a use of an adder tree in Figures 10, 11 and 12. Furthermore, Figure 14 illustrates an adder tree 220. Note that the adder tree 220 includes four full adders. None of these full adders are indicated as having inputs for winner-take-all bits or logic for processing winner-take-all bits as recited in claim 1.

At Col. 54, lines 34-54, Spiegel describes the adder tree 220 with the following words:

> "Reference is now made to FIG. 14, which illustrates the full adder tree 220 of FIG. 10, constructed and operative in accordance with a preferred embodiment of the present invention. As seen in FIG. 14, the full adder tree comprises a plurality of full adders based on commercially available integrated circuit full adders, such as the 74F283 commercially available from Philips, Eindhoven, Netherlands. In the present embodiment, the FIG. 14 full adder tree adds m=5 streams of input data."
>
> "Since 5 falls between the second and third powers of 2, a three-segmented full adder tree is provided, comprising four full adder trees. Suitable pipeline register elements 232 provide delays which "fill in" the full adder tree. The first segment adds two pairs of input data elements and provides the two sums to the full adder of the second segment, which adds them and provides the sum of the two sums to the full adder of the third segment. The fifth data element is delayed by the pipeline register elements 232 and is added to the sum of the two sums by the full adder of the third segment."
>
> "Analogous full adder trees can be constructed for other suitable values of m."

Note that this passage does not even remotely suggest that the adder tree 220 may receive and operate on winner-take-all bits as recited in claim 1.

The Examiner points to Figures 17, 26 and 27 of Spiegel as evidence for the anticipation of the winner-take-all processing feature of each adder cell as recited in claim 1. Applicant finds no evidence in any of these figures for an adder cell configured with the winner-take-all processing feature as recited in claim 1.

Figure 17 illustrates a serial-in parallel-out (SIPO) unit 240 that may be used as part of SIPO array 190 depicted in Figure 16 (Col. 30, lines 62-67). Figure 26 illustrates a method for controlling a two dimensional convolution, useful in implementing ALU 195 of Figure 12 (Col. 62, lines 25-27). Figure 27 illustrates a three dimensional convolution scaler which is an extension of the two dimensional convolution scaler of FIG. 12 (Col. 63, lines 20-23). None of these figures contain any teaching or suggestion concerning adder cells or anything similar to the winner-take-all processing feature.

Thus, claim 1 and its dependents are patentably distinguished over Spiegel at least for the reasons given above.

Claims 11, 20, 23 and 27 recite features similar to claim 1, and thus, these claims and their respective sets of dependents are patentably distinguished over Spiegel at least for the reasons given above in support of claim 1.

Claim 15 as herein amended recites:

> "A pixel computation unit comprising:
>
> a sample request unit configured to read samples from a sample buffer, and select one or more of the samples residing within one or more filter regions;
>
> one or more multiplication units configured to multiply a first sample component of each selected sample by a corresponding coefficient to generate one or more weighted first sample components;
>
> a first adder tree configured to receive the one or more coefficients used to obtain the one or more weighted first sample components, and to generate a coefficient sum comprising a sum of the one or more coefficients, wherein the first adder tree is further configured to receive the one or more weighted first sample components from the one or more multiplication units, and to generate a first summation of the weighted first sample components; and
>
> a division unit configured to divide the first summation by the coefficient sum to obtain a first pixel value."

This combination of features is not taught or suggested in Spiegel. In particular, Spiegel does not teach or suggest an adder tree configured to "receive the one or more coefficients used to obtain the one or more weighted first sample components, and to generate a coefficient sum comprising a sum of the one or more coefficients" as recited in claim 15.

At Col. 46, lines 50-57, Spiegel describes a 5x5 convolution computation as follows:

> "Specifically, the 5x5 convolution computation on the first "window" of CT pixels defined in the image p_{ij}, namely the first five pixels of the first five rows, or all p_{ij} for which i and j are both less or equal than 5, is as follows:
>
> Convolution result=1/25 x sum_j [$c_{.j}$ (sum_i (c_i. p_{ij}))]
>
> where summations over i and over j are both from 1 to 5."

There is no suggestion in this passage (nor anywhere else in Spiegel) that a summation of coefficients is performed as recited in claim 15.

Furthermore, Spiegel nowhere teaches or suggests "a sample request unit configured to read samples from a sample buffer, and select one or more of the samples residing within one or more **filter regions**".

Thus, claim 15 and its dependents are patentably distinguished over Spiegel.

CONCLUSION

Applicant submits the application is in condition for allowance, and an early notice to that effect is requested.

If any extensions of time (under 37 C.F.R. § 1.136) are necessary to prevent the above referenced application(s) from becoming abandoned, Applicant(s) hereby petition for such extensions. If any fees are due, the Commissioner is authorized to charge said fees to Meyertons, Hood, Kivlin, Kowert & Goetzel PC Deposit Account No. 50-1505/5181-56601/JCH.

Also enclosed herewith are the following items:

☒ Return Receipt Postcard

☒ Notice of Change of Address

□ Check in the amount of $ for fees ( ).

Respectfully submitted,

Mark K. Brightwell

Mark K. Brightwell
Reg. No. 47,446
AGENT FOR APPLICANT(S)

Meyertons, Hood, Kivlin, Kowert & Goetzel PC
P.O. Box 398
Austin, TX 78767-0398
Phone: (512) 853-8800
Date: May 26, 2004